UNIVERSITE MOHAMMED V RABAT-AGDAL

Faculté des Sciences

Département d'Informatique

SMI - Algo.II, 2014-2015

Série 2

Ex.1

1) Montrer que si T(n) est un polynôme de degré k alors $T(n) = O(n^k)$.
2) Montrer que, pour tout réel a, b (b >0) : $(n + a)^b = \Theta(n^b)$.

Ex.2

Prouver ou démentir les affirmations suivantes :

1) $2^{n+1} = O(2^n)$
2) $2^{2n} = O(2^n)$
3) $2^{2n} = O(n\,!)$

Ex.3

Classer les fonctions suivantes, selon l'ordre asymptotique grand-O :

$f_1(n) = 2^{1000000}$, $f_2(n) = n^2$, $f_3(n) = 1000000\, n$

$f_4(n) = n^{0.99999} \log n$, $f_5(n) = 1.000001^n$

Ex.4

Considérer deux algorithmes A1 et A2 avec leurs temps d'exécution respectifs

$T1(n) = 9\, n^2$ et $T2(n) = 100\, n + 96$.

1) En exprimant la complexité des deux algorithmes dans la notation grand-O, quel est le meilleur algorithme ?
2) Pour n = 10, votre choix d'algorithme est- il valide ?
3) Déterminer, à partir de quelle valeur de n, l'algorithme choisi est plus efficace que l'autre.
4) Quelle est la complexité de l'algorithme suivant, qui fait appel aux deux algorithmes A1 et A2 :

```
début
  A1 ;
  A2 ;
fin
```

Ex.5

Analyser, en utilisant grand-O, les algorithmes suivants :

```
A1(n)
début
 i :=1 ;  s := 0 ;
 tantque   i ≤ n  faire
        s := s + 1 ;
        i := 2 * i ;
 ftantque
fin
```

```
A2(n)
Début              s := 0 ;
 pour  i := 1  à  n  - 1  faire
   pour  j := i + 1  à  n  faire
        s := s + 1;
   fpour
 fpour
fin
```

```
A3(n)
début
 s := 0 ;
 pour   i := 1  à  n   faire
   s :=  s – 1 ;
   pour   j := 1  à  i   faire
        s := s + 2 ;
   fpour
 fpour
retourner(s) ;
fin
```

Quelle est la valeur calculée par A3 ?

Ex.6. On considère l'algorithme suivant, a et b sont des entiers strictement positifs tels que b≤ 2a :

```
Calcul(a,b)
début
        n := 0 ;   m := b ;
        Tantque    m ≤  a   faire
                m  :=  2 * m ;
                n :=  n + 1 ;
        ftantque
  retourner(n) ;
fin
```

1) Montrer que la condition suivante reste vraie avant, à chaque itération, et après l'exécution de tantque : $n \geq 0, \quad m = 2^n b \quad \text{et} \quad m \leq 2a$
2) En déduire que $n = E(\log_2 (a / b)) + 1$

Ex.7

Etant donnés deux tableaux T1[1..n] et T2[1..n], chaque tableau Ti contient les chiffres d'un entier positif n_i, (le chiffre des unités est à la première position, celui des dizaines à l'indice 2, etc …). On suppose que les deux entiers n_1 et n_2 ont un même nombre de chiffres n (sinon on complétera le tableau du plus petit par des zéros).

1) Ecrire un algorithme qui fait la somme, chiffre à chiffre, de deux entiers positifs n1 et n2, représentés respectivement par T1 et T2. Le résultat est un tableau T[1..n+1]. Donner sa complexité.
2) Evaluer la complexité du produit de n1 et n2, représentés de la même façon que précédemment.

Ex.8 Reprendre la série1 et évaluer la complexité de chaque algorithme.